[Title]

Data transmission record keeping method

5   [Field of the invention]

The present application relates to a method of keeping a
record of a data transmission, to a method of controlling a
sending peer of a data unit transmission protocol and to a
10   corresponding sending peer, as well as to a method of
controlling a receiving peer of a data unit transmission
protocol and to a corresponding receiving peer.

[Background of the invention]
15

The present invention relates to the field of communication
via data unit transmission. In data unit communication, a
stream of data symbols (e.g. bits or bytes) is divided by a
sender into units, sent over an appropriate transmission
20   network to a receiver, and the receiver reconstructs the data
symbol stream on the basis of the received data units.
Usually, this is performed in the context of a layering
scheme (such as the OSI scheme), i.e. where a sending peer of
a given protocol receives a data symbol stream from a higher
25   layer, generates data units in accordance with the given
protocol to which the sending peer adheres, and passes the
generated data units to lower layers. On the receiving side,
the receiving peer receives the data units from the lower
layer, reconstructs the data symbol stream, and passes the
30   reconstructed data symbol stream to the higher layer. The
concept of dividing a data symbol stream into data units and
the concept of layering are well known in the art, such that
a further explanation is not necessary here.

35   It is noted that such sub-divisions of data are referred to
by a variety of names, such as packets, segments, frames,
protocol data units, service data units, etc. In the context

of the present specification and claims, the term "data unit"
is used generically to relate to any such sub-divisions of
data for the purpose of communication.

5   Fig. 5 is a schematic representation of a sending and a
receiving peer of a communication. More specifically, two
peers 51 and 52 of a predetermined layer LX are shown. The
layer LX can e.g. be the link layer L2. For the purpose of
the present explanation it will be assumed that peer 51 is
10   acting as sending peer and peer 52 as receiving peer. In this
case sending peer 51 receives a stream of data symbols from
higher layer LX+1. Sending peer 51 comprises a data unit
generating part 510, e.g. a buffer, and a control part 511,
e.g. a processor for controlling the data unit generating
15   part 510. Furthermore, it is assumed that a plurality of
channels 53 for lower layer LX-1 exist, and that sending peer
51 comprises a switching part 512 for selecting a channel,
where control part 511 controls the data unit generator 510
in such a way that data units of the correct format are
20   generated for a given channel. In other words, each of the
channels is associated with a particular data unit format.
The lower layer LX-1 (and possibly layers below LX-1) provide
a connection 55 to receiving peer 52, where the data units
arrive over channels 54 that correspond to the channels 53.
25   Consequently receiving peer 52 has a switching part 522 for
receiving data units from the appropriate channel 54, and the
data units are processed in a part 520 that serves to
reconstruct the data symbol stream under control of a control
part 521. The reconstructed data symbol stream is passed to
30   higher layer LX+1.

For example, if LX is the link layer L2, then the various
channels 53, 54 can e.g. relate to a variety of different
physical communication paths, such as a variety of WCDMA
35   (Wideband Code Division Multiple Access) connections, a HSDPA
(High-Speed Down Link Packet Access) connection, a GSM
(Global System for Mobile Communication) connection, a WLAN

(Wireless Local Area Network) connection, or some other
wireless type of connection.

In order for the receiving peer 52 to be able to reconstruct
5   the data symbol stream, each data unit carries a sequence
position indicator that indicates a position in a sequence of
data units. By correctly ordering the sequence in accordance
with the sequence position indicators, the receiving peer can
reconstruct the data symbol stream.
10
In the scenario of Fig. 5, it is assumed that at least some
of the channels 53, 54 require data units of a different
format (e.g. of a different size).

15   As a consequence, if the sending peer 51 is in the process of
sending data units over one of the channels 53 in accordance
with a format associated with said channel, then it keeps a
record of the amount of sent data symbols of said data symbol
streams in terms of the data units of the format being used.
20   In other words, the sending peer keeps a record of a sequence
of data units in terms of the sequence position indicators
where the records shows which data units have been sent and
which have not.

25   In the situation as indicated in Fig. 5, the following can
occur. It can become necessary or desirable to switch from a
given channel (and given data unit format) to a new channel
(and therefore possibly new data unit format) during the
ongoing transmission of a data symbol stream. In order to
30   guarantee a correct reconstruction of the data symbol stream
at the receiving peer, it is envisionable to purge the
sending buffer in data unit generator 510 together with all
of the corresponding records kept in terms of the initial
data unit format and corresponding sequence position
35   indicators, and to restart sending with the new data units in
their new format, where the corresponding new sequence

position indicators are also reset, i.e. start at the initial value (zero or one).

5   Such a procedure for purging a send buffer and resetting the corresponding records may lead to a number of disadvantages. For one thing, the purging can lead to a loss of parts of the data symbol stream. Also, if ARQ (Automatic Repeat Request) is implemented between the sending peer 51 and receiving peer 52, then there the problem that after a switch in channels

10 and in formats, it is difficult to appropriately retransmit lost or damaged data units.

[Object of the present application]

15 It is the object of the present application to provide an improved method of managing data units in a system where data units can be sent in at least a first format and a second format from a sending peer to a receiving peer.

20 [Summary of the invention]

The object is solved by the methods and devices described in the independent claims. Advantageous embodiments are described in the dependent claims.

25

In accordance with the present invention, a sending peer of a data unit transmission protocol that is capable of dividing a data symbol stream into data units of at least a first format and a second format maintains a first record of the data

30 symbol stream in terms of one or more first sequences of data units of the first format and simultaneously maintains a second record of the data symbol stream in terms of one or more second sequences of data units of the second format. The first record and second record have a common reference point

35 to the data symbol stream, e.g. the beginning of the respective first data unit.

Due to the simultaneous record keeping it is not necessary to
purge a sending buffer or reset records when switching from
the first format to the second format. The sending peer is
capable of dynamically switching between a first transmission
mode for sending data units in the first format and a second
transmission mode for sending data units in the second
format, where it is not necessary to reset the sequence
position indicators after the switching in transmission mode.
Namely, after switching from the first transmission to the
second transmission mode, the transmission continues with a
given data unit of one of the second sequences comprising a
position indicator such that the given data unit comprises a
data symbol immediately following the last data symbol of the
data symbol stream that was sent in the data unit of the
first format last sent before the switching. Equally, after
switching from the second transmission mode to the first
transmission mode, the transmission continues with a given
data unit of one of the first sequences comprising a position
indicator such that the given data unit comprises a data
symbol immediately following the last data symbol of the data
symbol stream that was sent in the data unit of the second
format last sent before the switching.

It is noted that there can naturally be more than two
formats, in which case the sending peer may keep respective
records for each format.

The receiving peer of the present invention is capable of
identifying the respective sequence position indicator
associated with the first sequence of the first format or
second sequence of the second format, and is capable of
detecting the switching of the sending peer between the first
transmission mode and second transmission mode. The
reconstructing of the data symbol stream occurs on the basis
of the identified sequence position indicators and the
detected switching.

It is noted that the receiving peer does not necessarily have
to keep a first and a second record respectively associated
with the first sequence of data units of the first format and
second sequence of data units of the second format, using a
5   common reference point to the data symbol stream. Namely, it
is possible that the receiving peer reconstructs the data
symbol stream using dedicated instruction messages from the
sending peer. In other words, the sending peer sends
instruction messages to the receiving peer, where the
10   instruction messages tell the receiving peer how to assemble
the data symbol stream on the basis of the data symbols of
the first and second format and the associated sequence
position indicators of the respective formats.

15   However, it is preferable that the receiving peer maintains a
first record associated with the data units of the first
format and a second record associated with the data units of
the second format, and reconstructs the data symbol stream on
the basis of a common reference point of the first sequence
20   of data units of the first format and the second sequence of
data units of the second format.

[Brief description of figures]

25   In the following examples of the present invention will be
described on the basis of detailed embodiments, which make
reference to figures, where:

Fig. 1   schematically shows an example of a first sequence
30   of data units of a first format and a second
            sequence of data units of a second format,

Fig. 2   schematically shows an example of sent data units
            and received data units,

35

Fig. 3   schematically shows three first sequences of data
            units of the first format and two second sequences

of data units of the second format, where there are respective offsets between the data units of the first sequences and the second sequences,

Fig. 4   schematically shows an example where the data unit size is variable,

Fig. 5   schematically shows a sending peer and receiving peer, in which the concepts of the present invention can be implemented,

Fig. 6   shows a flow chart of an embodiment of the present invention,

Fig. 7   shows a flow chart of another embodiment of the invention, and

Fig. 8   shows a flow chart of a further embodiment of the invention.

[Detailed description of embodiments]

Fig. 1 schematically shows an example of a method of keeping a record of a data transmission in a sending peer in accordance with the present invention. Arrow 10 represents a data symbol stream. The data symbols can be present in any suitable or desirable way, e.g. can be bits, bytes, etc. The data symbol stream can already have a certain structure as provided from the higher layer, e.g. can be divided into higher layer data units. This is indicated by sequence 11 of data units a, b, c. For example, if the data units being generated belongs to the link layer L2, then the data units a, b, c of sequence 11 could be protocol data units (PDUs) of layer L3.

In accordance with the present embodiment of the invention, a sending peer of a data unit transmission protocol that is

transmitting data symbol stream 10 maintains a first record
of the data symbol stream 10 in terms of a first sequence 12
of data units A1-A11 of a first format A, and simultaneously
maintains a second record of the data symbol stream 10 in
5   terms of a second sequence 13 of data units B1-B15 of a
second format B. The first and second record have a common
reference point 14 to the data symbol stream 10.

An example of such a method is shown in Fig. 6. Step S61
10 represents the procedure for controlling the transmission of
data units, and steps S62 and S63 represent the simultaneous
maintenance of corresponding records with respect to the
associated formats, regardless of which format was used in
the actual sending. Naturally, the procedure of Fig. 6 is
15 only an example, and other embodiments are possible, e.g.
steps S62 and S63 may also be conducted sequentially, as this
still leads to a simultaneous record keeping with respect to
the two formats. Although not shown in Fig. 6, the keeping of
more than two records for corresponding more than two formats
20 is also possible and straight-forwardly achieved.

Each sequence 12, 13 has sequence position indicators for
identifying the correct position in the respective sequence.
In Fig. 1 these sequence position indicators are shown as
25 simple numbering, together with an indication of the
respective format. Therefore, e.g. "A1" represents the data
unit of format A having the sequence position 1, "B3"
represents the data unit of format B having the sequence
position 3, etc.
30
It is noted that the data units A1-A11 and B1-B15 represent
the payload of actual data units. However, as shall be
explained in the following with reference to Fig. 2, not
every shown data unit is actually sent. Much rather, Fig. 1
35 signifies that a record is simultaneously kept in terms of
both shown formats, in order to allow a dynamic switching

between sending data units of format A from sequence 12 or data units of format B from sequence 13.

Fig. 1 is only a schematic representation, and the record can
5   be kept in any suitable or desirable way. For example, the record can be provided by a mapping between data symbol positions and the respective sequence position indicators. The sequence position indicators can e.g. be simple whole numbers, as indicated in Fig. 1. However, there are also
10 other possibilities. For example, the data symbol positions can themselves be used as sequence position indicators. Namely, the last data symbol position in a data unit can be used for identifying the data unit. For example, if one assumes that the data units of format A have a length of N
15 data symbols, then (assuming a starting point of data symbol number 1) A1 can be identified by data symbol number N, A2 by data symbol number 2N, etc.

All records can be kept in the same fashion, or one record
20 can be kept by means of a conversion function from another record. For example, when using natural numbers as sequence position indicators, the sequence position indicator for format A can e.g. be determined from the sequence position indicators for format B by:
25

$$No(A) = \left\lfloor No(B) \cdot \frac{Len(B)}{Len(A)} \right\rfloor + 1$$

where No(A) indicates the sequence position number for format
A, $\lfloor\ \rfloor$ indicates an integer truncation, No(B) indicates the
30 sequence position number for format B, Len(A) indicates the length of data units of format A, and Len(B) indicates the length of data units of format B.

In some data transmission systems the sequence position
35 numbering space is limited. In such cases, a so-called wrap-around may occur. For example, if the sequence position is

given by 10 bits, then a wrap-around will occur every 1024 positions. In general, this problem can be solved by employing an appropriate modulo operation in the addressing of a given data symbol position in terms of a specific data
5   unit format, and equally using modulo operations when converting from sequence position indicators in one format to sequence position indicators in another format.

In the following, a specific example of record keeping in
10   such a situation will be explained. For this example, a constant data unit length is assumed. It will also take into account the possibility of a wrap-around occurring in the sequence position numbering.

15   In Fig. 1, the reference s represents a data symbol position in stream 10. In terms of the first sequence 12 of data units of format A and second sequence 13 of data units of format b, the symbol position s can be represented or addressed as {$SN_A(s)$; $Pos_A(s)$} in terms of format A and as {$SN_B(s)$;
20   $Pos_B(s)$} in terms of format B. $SN_A(s)$ represents the sequence position of the corresponding data unit in the first sequence of format A, $SN_B(s)$ represents the sequence position of the corresponding data unit in the second sequence of format B, $Pos_A(s)$ represents the position of s within the data unit of
25   format A and $Pos_B(s)$ represents the position of s within the data unit of format B. In the example of Fig. 1, $SN_A(s)$ would be A4, $SN_B(s)$ would be B5, and arrow 15 represents $Pos_A(s)$ and arrow 16 represents $Pos_B(s)$.

30   The representation of s in a format X, i.e. {$SN_X(s)$; $Pos_X(s)$}, can be determined as:

$$SN_X = \left\lfloor \frac{s \bmod (MaxSN(X) \cdot Len(X))}{Len(X)} \right\rfloor$$

35

$$Pos_X = [s \bmod (MaxSN(X) \cdot Len(X)] \bmod Len(X)$$

where the reference point is assumed as shown in Fig. 1, i.e.
$\{SN_X(s=0)=0; Pos_X(s=0)=0\}$.

5    $\lfloor \rfloor$ represents an integer truncation and a mod b represents a
modulo operation. MaxSN(X) is the maximum of the limited
sequence position numbering space, and Len(X) is the length
of a data unit (more specifically the length of the payload
of a data unit) of format A.

10

In addition, for each format X, a reference value $Ref_X(s)$ can
be determined, which specifies the effect the offset of the
representation window start and the reference point:

15

$$\text{Ref}_X(s) = MaxSN(x) \cdot Len(X) \cdot \left\lfloor \frac{s}{MaxSN(X) \cdot Len(X)} \right\rfloor$$

An arbitrary format A can then be converted into an arbitrary
format B as follows:

20

$$SN_B(s) = \left\lfloor \frac{(SN_A(s) \cdot Len(A) + Pos_A(s) + \Delta(s))\mod(MaxSN(B) \cdot Len(B))}{Len(B)} \right\rfloor$$

$$Pos_B(s) = \left[(SN_A(s) \cdot Len(A) + Pos_A(s) + \Delta(s))\mod(MaxSN(B) \cdot Len(B))\right]\mod Len(B)$$

where $\Delta(s)$ = $Ref_A(s)$ - $Ref_B(s)$.

25

Fig. 2 schematically shows a sequence 20 of data units sent
by the sending peer and a sequence 21 of data units received
by a receiving peer.

30   In the example of Fig. 2, the transmission starts with data
units of the first format A. The sender sends data units 1A-
4A. In order to distinguish the record as shown in Fig. 1
from the actual data units sent, the data units of Fig. 2 are
identified by the sequence position indicator (1, 2, ...)
35   following by the format indicator (A, B), as opposed to the

reference in Fig. 1, where the sequence position indicator
follows the format indicator.

In the example of Fig. 2, at point 22 a mode switch occurs in
5   the sending peer. Namely, the sending peer dynamically
switches from the first transmission mode A to the second
transmission mode B. There can be a variety of reasons for
switching the transmission mode e.g. due to changes in the
availability of channels for transporting the data units of
10 the first format A or the second format B. In any case, the
present invention is not concerned with the reasons for
performing a format switch.

After switching from the first transmission mode A to the
15 second transmission B, the transmission continues with data
unit 6B that corresponds to B6 of second sequence 13, because
B6 contains the data symbol immediately following the last
symbol of the data symbol stream 10 that was sent in data
unit 4A before the mode switch 22.
20
The transmission therefore continues with the data units 6B-
9B of second format B, until a further mode switch 23 occurs,
back to the first format A. After switching from the second
transmission mode to the first transmission mode, the
25 transmission continues with data unit 7A that corresponds to
A7 of the first sequence 12, because A7 comprises the data
symbol immediately following the last data symbol of the data
symbol stream 10 that was sent in data unit 9B before the
mode switch 23.
30
In order to identify the correct data unit after a mode
switch, the first sequence 12 and second sequence 13 share a
common reference point 14 to the data symbol stream 10. In
the example of Fig. 1, the common reference point is the
35 beginning of the data symbol stream, i.e. the first data
symbol, where the first data symbol of the data symbol stream
is assigned as the first symbol of the data unit carrying the

first sequence position indicator in both sequences 12 and
13.

The setting of a reference point in this way is the
5   intuitively most appealing, and therefore preferred. However,
the present invention is by no means restricted to such a
reference point, as the reference point common to the
sequences and the data symbol stream can be chosen
arbitrarily and can be adapted dynamically.
10
Also, it is noted that the term "data symbol stream" relates
to any stream of data symbols, i.e. to a complete amount of
data for sending or to a contiguous subset of such a complete
amount. In other words, in an overall stream of data symbols,
15   the beginning and end for defining the "data symbol stream"
can be chosen arbitrarily.

Fig. 7 shows a flow chart of an example of the above
described control method. In step S71 it is determined
20   whether a mode switch should occur. If no, the process
proceeds to step S72, where the sending of data units
continues with the present format. On the other hand, if step
S71 indicates a change in mode and thereby in format, then
the correct sequence position indicator for the new format is
25   determined in the above described way, and then a data unit
of the new format and having the determined sequence
indicator is transmitted.

Returning to Fig. 2, the receiving peer receives the sequence
30   21 of data units of both formats A and B. In order to
reconstruct the data symbol stream, the receiving peer
identifies the respective sequence position indicators in
each received data unit and detects a switching of the
sending peer from the first format A to the second format B,
35   as at switch 22, or from the second format B to the first
format A, as at switch 23.

An example of this is shown in Fig. 8. The shown method is
conducted after receiving a data unit. In step S81 a switch
in transmission mode is detected. Then, in step S82 the
sequence position indicator is identified. Finally, in step

5   S83, the reconstruction is performed in dependence on the
identified sequence position indicator and on the possibly
detected mode switch. Naturally this is only an example, as
steps S81 and S82 could also be executed concurrently, or S82
could be executed before step S81.

10

In order to correctly reconstruct the data symbol stream, the
receiving peer is capable of determining how to assemble the
data units of the first format A and the second format B at
the transmission of a mode switch. This detection can be done

15   in a number of ways. For one thing, it is possible that the
sender sends dedicated messages indicating a switch in data
unit format, where said messages may also indicate how to
assemble the last data unit before the switch and the first
data unit after the switch.

20

Preferably, the receiving peer also maintains a first and a
second sequence, similar to what is shown in Fig. 1, using a
common reference point to the data symbol stream. The common
reference point can be implicitly defined with respect to the

25   sequence position indicators as e.g. shown in Fig. 1. Namely,
it can be a common rule known to all sending peers and
receiving peers that the reference point is the first data
symbol contained in the respective date unit carrying the
first sequence position indicator.

30

However, it is equally possible that the sending peer sends a
reference point synchronization message to the receiving peer
for setting the reference point, where the receiving peer
sets its common reference point on the basis of the reference

35   point synchronization method. The synchronization message can
be used for an initial setting or for a subsequent resetting
of the reference point.

The change in format at the point in time of a mode switch
can be communicated in different ways from the sending peer
to the receiving peer. For example, the different formats can
5   be fixedly associated with given channels, as already
explained in connection with Fig. 5. In other words, the
format is identified by the channel over which the data unit
is sent and received. As such, the changing of one channel to
another serves to identify the switch in transmission mode by
10   the sending peer.

On the other hand, it is alternatively or additionally
possible for the sending peer to send a message indicating an
associated switching in data unit format.
15
As a further alternative or additional possibility, it is
possible to send a format type indicator in each data unit.
The receiving peer is then arranged to identify the format
type indicator from each received data unit, where a change
20   in format and correspondingly a switch in the transmission
mode can straight-forwardly be recognised.

Attention is also drawn to the fact that the sequence
position indicator provided in the data units can be chosen
25   in any desirable or suitable form. For example, it can be in
the form indicated in Figs. 1 and 2, i.e. a consecutive whole
number count. However, the sequence position indicator can
also be of the type known from TCP/IP, i.e. can be an
indication of the number of data symbols that has been
30   transported. In other words, the sequence position indicator
can be a count value that directly relates to the data unit
or a count value that relates to the contents of the data
unit.

35   As explained above, the receiving peer is able to reconstruct
the data symbol stream on the basis of the identified
sequence position indicators and the detected mode switching.

When reconstructing the data symbol stream, the receiving
peer eliminates the duplicate data symbols that where
unnecessarily sent as a cause of the mode switching. This can
be explained with reference to boxes 24 and 25 in Fig. 2.
5   Namely, when switching from format A to format B at point 22,
the sending of data unit 6B leads to a renewed sending of the
data symbols identified by box 24 as an overlap between data
units A4 and B6 of Fig. 1. Equally, when switching from
format B to format A, at point 23, the data symbols indicated
10   by box 25 are sent twice, as an overlap between data unit B9
and A7 of Fig. 1.

As indicated above, the receiving peer can e.g. identify the
redundant data symbols on the basis of dedicated messages
15   from the sending peer, which dedicated messages e. g.
identify specifically which part of data unit 6B or 7A to
eliminate (e.g. using the values $Pos_x(s)$ described above) ,
or on the basis of the common reference point between the
first sequence 12 and the second sequence 13, which common
20   reference point allows to identify the respective overlap
between the two sequences 12, 13.

Nonetheless, as can be seen from the above described example,
the performance of a mode switch and the subsequent change in
25   data unit format can lead to a considerable amount of
redundant or duplicate data transmission. In order to avoid
such a redundant transmission of data symbols, the present
invention is preferably arranged in a way that will be
explained in connection with Fig. 3. As can be seen in Fig.
30   3, data symbol stream 30, which again may be received in the
form of higher layer data units a, b, c, in sequence 31, is
not only recorded in terms of a first sequence and a second
sequence, but in terms of a plurality of first sequences 32,
$32+\Delta$, $32+2\Delta$ and/or a plurality of second sequences 33, $33+\delta$.
35   Δ and δ represent respective offsets of the associated
sequence with respect to the common reference point. For
example, the offsets Δ and δ can be expressed in terms of

data symbol positions, e.g. they can be equal to one data
symbol position, two data symbol positions etc.

By keeping one or more first sequences 32 and/or one or more

5   second sequences 33, the amount of redundant data symbols
transmitted after a mode switch can be reduced in the
following way. After switching from the first transmission
mode A to the second transmission mode B, the sending peer
determines the given data unit of the first of the second

10   sequences, i.e. sequence 33, that comprises the data symbol
immediately following the last data symbol that was sent in
the last sent data unit of the first format A, and determines
the given data unit of the second of the second sequences,
i.e. 33+δ, that comprises the data symbol immediately

15   following the last data symbol that was sent in the last sent
data unit of the first format A. The sending peer then
determines which of the given data units comprises less
duplicate data symbols already sent with the last data unit
of the first format A, and continues the transmission with

20   that one sequence of said second sequences 33, 33+δ to which
the given data unit was less of said data symbols belongs.

Naturally, the same process can be conducted in the inverse
direction, i.e. when switching from format B to format A and

25   then identifying which one of the sequences 32, 32+Δ, 32+2Δ
contains the data unit that has the smallest overlap with
respect to the previously sent data unit of the previous
format.

30   The offset can be communicated to the receiving peer in any
suitable or desirable way. For example, it can be sent in a
dedicated message that indicates that all following data
units have a certain offset. Alternatively or additionally
the offset may be communicated by an offset indicator

35   contained in each data unit that is sent. The receiving peer
can then read the offset indicator and make an appropriate
adjustment with respect to the common reference point.

If the data unit transmission protocol to which the sending
peer and receiving peer adhere provides for the sending of
receiver status messages from the receiving peer to the
5   sending peer, then the present invention may be embodied in
the following way. The receiver status messages comprise
information on the receipt of one or more of the data units,
where the data units are identified in the messages in terms
of sequence position indicators associated with a given
10   sequence, e.g. the first sequence 12 of Fig. 1, or one or
more of the first sequences 32, 32+Δ, 32+2Δ as shown in Fig.
3.

The receiver status messages can be chosen in any suitable or
15   desirable form, can e.g. be acknowledgement messages, non-
acknowledgment messages, etc.

The sending peer comprises a retransmission procedure for
retransmitting data units on the basis of the receiver status
20   messages. It furthermore implements a procedure for
determining one or more of the second sequence position
indicators B1 - B15, B1+δ - B15+δ associated with one of the
second sequences 33, 33+δ on the basis of a given first
sequence position indicator. The one or more second format
25   data units cover all of the data symbols contained in the
given data unit of the first format associated with the first
sequence position indicator. For example, looking at the
example of Fig. 3, the sending peer is able to identify data
units B5, B6 or B4+δ, B5+δ as containing all the data symbols
30   of data unit A4 of format A. Then, the sending peer is
capable of performing a retransmission procedure that
retransmits the data units of the second format B associated
with the identified second sequence position indicators.

35   In other words, if the sending peer would like to retransmit
the data unit carrying A4, e.g. because A4 is identified as
defective in a receiver status message or has possibly caused

a time-out, then the sending peer is capable of identifying
the second format data units, e.g. B5, B6 that can be
retransmitted in order to transport the same data symbols as
contained in A4.

5

Naturally, this process is also possible in the inverse
direction, i.e. identifying one or more sequence indicators
of a first sequence 32, 32+Δ, 32+2Δ on the basis of a given
sequence position indicator associated with one of the second
10   sequences 33, 33+δ. An example of this is shown in Fig. 2,
where arrow 28 at the bottom represents a data retransmission
period. Moreover, the hatching of data units 2A and 7B on the
side of receiving peer 21 symbolize that these where not
correctly received and that a corresponding receiver status
15   message (not shown in Fig. 2) was sent to the sending peer.
In the example of Fig. 2, the retransmission period 28 occurs
during a time where the sending peer is in a transmission
mode for sending data units of the first format A. As a
consequence, the defective data unit 2A is simply
20   retransmitted as it is. However, the sending peer determines
the sequence position indicators of the first sequence 12 of
Fig. 1, associated with defective data unit 7B of the second
format B. These data units are A5, A6, such that the sending
peer retransmits data units 5A and 6A in order to transport
25   the data symbols contained in 7B. As indicated by boxes 26
and 27, this can lead to a certain amount of duplicate data
symbols being sent. The amount of duplicate data symbols sent
can be reduced by using the above described offset principle.

30   In correspondence to the sending peer, a receiving peer of
the present invention is preferably arranged such that it can
determine second sequence position indicators on the basis of
a first sequence position indicator, in order to generate
appropriate receiver status messages. More specifically, the
35   receiving peer preferably implements a procedure for
determining one or more second sequence position indicators
B1 - B15, B1+δ - B15+δ associated with one of the second

sequences 33, 33+δ on the basis of a first sequence position
indicator associated with one of the first sequences 32,
32+Δ, 32+2Δ, where the determined one or more data units of
the second format B associated with the determined one or
more second sequence position indicators cover all of the
data symbols contained in the given data unit of the first
format A associated with the first sequence position
indicator. Then, the data unit of the first format A can be
identified in terms of the one or more second sequence
position indicators in a receiver status message.

In this way, it is possible for the receiving peer to always
identify data units in receiver status messages in accordance
with one format. Namely, if it receives the data units in
that one format, then the identification is straight-forward
on the basis of the sequence position indicator contained in
that data unit, and otherwise the above-described procedure
is used for determining the sequence position indicator from
data units of another format. For example, using the
embodiment of Fig. 1, the receiving peer could generate
receiver status messages that indicate received data symbols
in terms of sequence position indicators associated with
sequence 12, even if data units of format B have been
received.

Preferably, the receiving peer is capable of performing the
determination in both directions, i.e. being able to
determine one or more first sequence position indicators
associated with one of the first sequences 12, 32, 32+Δ,
32+2Δ on the basis of a second sequence position indicator
B1-B15, and determining one or more second sequence position
indicators associated with one of the second sequences 33,
33+δ on the basis of a first sequence position indicator A1-
A11. In other words, the receiving peer is preferably
arranged in such a way that it can identify data symbols sent
in a data unit of format A with sequence position indicators
associated with data units of format B, and vice versa. This

capacity is preferably combined with a procedure for
generating received status messages comprising one or both of
first and second sequence position indicators depending on
one or more predetermined optimisation functions.

Namely, the receiving peer can select which representation of
data it wants to use in a receiver status message. It can
choose to use the representation with respect to one format,
the other format, or to mix the representation with respect
to both formats. As an example, when looking at the received
data units of sequence 21 in Fig. 2, the receiving peer can
identify data unit 2A in terms of format A (as 2A), or in
terms of format B (as B2, B3). Equally, the receiving peer
can identify data unit 7B in terms of format B (i.e. as 7B)
or in terms of format A (as A5, A6).

The one or more optimisation functions to be used can be
chosen in any suitable or desirable way. For example, the
optimisation function can be used to identify data units in
the receiver status messages in terms of the format presently
being used by the sending peer. In other words, the receiving
peer determines that the sending peer has a transmission mode
that sends format A, all data units are identified in the
receiver status messages in terms of format A.

A further example of an optimisation function is to optimise
the size of the receiver status messages. Namely, in such a
case the receiving peer is arranged to represent received
data units in terms of the format that allows the shortest
identification of data units, i.e. a format that corresponds
to data units of large size, such that the receiver status
messages become short. This can have the disadvantage that an
increased amount of duplicate data symbols are transmitted,
because the granularity of identifying data units is very
coarse, because the receiver status messages are intended to
be short and thereby only very coarsely identify the data
units.

As a consequence, the optimisation function can also be chosen in the exactly opposite way, i.e. to allow the finest granularity, which means that the data·units are expressed
5   with respect to the format of smallest size.

Now two special cases will be discussed. One special case is that the data symbol stream buffer and the sending peer runs empty for a certain period, such that the last data unit of a
10 given format is partly filled with padding. In order to allow the receiving peer to correctly re-construct the data symbol stream, it is capable of identifying the padding (e.g. on the basis of information in the data unit header that specifies the length and position of the padding) and a certain
15 synchronization rule. The synchronization for determining how the first data symbol of new data is addressed can be implicit, e.g. that the next data symbol is represented in the next data unit of each respective format. In other words, if padding would e.g. be contained at the bottom of data unit
20 A5 of Fig. 1, then the receiver could identify B6 and B7 as corresponding to A5, such that the next data unit of format A is A6 and the next data unit of format B is B8, such that the implicit resynchronisation would occur with respect to the beginning of A6 and B8. Naturally, it is also possible that
25 the sending peer sends an explicit reference point setting ·message to the receiving peer, which in the above example would e.g. identify data units A6 and B8.

A second special case will be described with respect of Fig.
30 4. Fig. 4 shows a case where the data units of a given format (more specifically the payload of said data units) has a varying size. As an example, there may be an optional header field that is only used in certain cases and can have a variable length, such that the payload within a data unit is
35 equally variable in order to accommodate the additional header field. An example of such a variable length header field is the length indicator field in the WCDMA RLC (radio

Link Control) protocol. Consequently, the reference
synchronization between format A and format B requires
keeping track of the data unit size. At the side of the
sending peer this is always known, and therefore no problem.

5   However, at the side of the receiving peer this information
is only available if the data units have been received
correctly. If data units get lost or are damaged, then this
information is not known to the receiving peer. Such a
situation is shown in Fig. 4, where the data units of format

10 A have a varying size, and it is assumed that data units 2A,
3A, 4A were not correctly received by the receiver in
sequence 42.

This problem can be solved by the use of the reference point

15 synchronization message already mentioned earlier. In the
case of Fig. 4, the reference point synchronization message
43 is sent at the point of time of a mode switch. The
reference point synchronization message contains information
on the change in reference point due to the variable data

20 unit size. As an example, if the size of data units 1A and 4A
is the standard size, and data units 2A and 3A are
respectively decreased by predetermined factors, then the
reference point synchronization message 43 could simply
indicate the amount of size reduction in data units 2A and

25 3A.

The above-described features of the invention can be
implemented in data unit transmission devices that act as
sending peers and/or receiving peers in the form of hardware,

30 software or any suitable combination of hardware and
software. According to a preferred embodiment, the present
invention is embodied in the form of a computer program
executable on a data unit transmission device.

35 The present invention can especially be implemented in the
context of the sending peer 51 and receiving peer 52 shown in
Fig. 5. More specifically, the control part 511 can be

arranged to operate as a record keeping part that simultaneously maintains the above-mentioned first and second record. Element 512 represents a switching part for dynamically switching between the above-described first and
5   second transmission mode, and the data unit generating part 510 can be arranged to act a data unit output part that is equipped such that after the switching part 512 switches from a first transmission to a second transmission mode the output part 510 continues the transmission with a given data unit of
10   one of the second sequences comprising a position indicator such that the given data unit comprises a data symbol immediately following the last data symbol of the data symbol stream that was sent in the data unit of said first format last sent before said switching, and that after the switching
15   part 512 switches from the second transmission mode to the first transmission mode, the output part 510 continues the transmission with a given data unit of one of the first sequences comprising a position indicator such that the given data unit comprises a data symbol immediately following the
20   last data symbol of the data symbol stream that was sent in the data unit of the second format last sent before the switching.

The control part 511 can especially be arranged to act as an
25   information processor for determining data symbols of format A in terms of sequence position identifiers associated with format B and vice versa.

Equally, the control part 521 of receiving peer 52 can act as
30   a sequence position identifier for identifying sequence position indicators in received data units, as a switching detector for detecting a switch in transmission mode at the sending peer. The element 520 of receiving peer 52 acts as a data symbol stream reconstruction part for reconstructing the
35   data symbol stream on the basis of the sequence position indicators and the switching detected by control part 521. The control part 521 may equally act as a message generator

for generating and sending receiver status messages to the
sending peer.

Although the present invention has been described with
respect to detailed embodiments, it is by no means restricted
thereto, as the invention is defined by the appended claims.
Furthermore, reference numerals in the claims serve to make
the claims easier to read, but are not intended to have any
limiting effect.